Rec'd PCT/PTO 08 APR 2005
PCT/AU03/01298

REC'D 27 OCT 2003
WIPO PCT

AUSTRALIA

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN COMPLIANCE WITH RULE 17.1(a) OR (b)

Patent Office
Canberra

I, JONNE YABSLEY, TEAM LEADER EXAMINATION SUPPORT AND SALES hereby certify that annexed is a true copy of the Provisional specification in connection with Application No. 2003901021 for a patent by BHP BILLITON INNOVATION PTY LTD as filed on 05 March 2003.

COMMONWEALTH OF AUSTRALIA
AUSTRALIA
PATENT OFFICE

WITNESS my hand this
Thirteenth day of October 2003

J R Yabsley

JONNE YABSLEY
TEAM LEADER EXAMINATION
SUPPORT AND SALES

SYSTEM AND METHOD(S) OF MINE PLANNING, DESIGN AND PROCESSING

FIELD OF INVENTION

The present invention relates to the field of extracting resource(s) from a particular location. In one form, the present invention relates to the planning, design and processing as applicable to a mine location in a manner based on enhancing the extraction of material considered of value, relative to the effort and/or time in extracting that material.

BACKGROUND ART

In the mining industry, once material of value, such as ore situated below the surface of the ground, have been discovered, there exists a need to extract that material from the ground.

In the past, a more traditional method has been to use a relatively large open cut mining technique, whereby a great volume of waste material is removed from the mine site in order for the miners to reach the material considered of value. For example, referring to Figure 1, the mine 101 is shown with its valuable material 102 situated at a distance below the ground surface 103. In the past, most of the (waste) material 104 had to be removed so that the valuable material 102 could be exposed and extracted from the mine 101. In the past, this waste material was removed in a series of progressive layers 105, which are ever diminishing in area, until the valuable material 102 was exposed for extraction. This is not considered to be an efficient mining process, as a great deal of waste material must be removed, stored and returned at a later time to the mine site 101, in order to extract the valuable material 102. It is desirable to reduce the volume of waste material that must be removed prior to extracting the valuable material.

The open cut method exemplified in Figure 1 is viewed as particularly inefficient where the valuable resource is located to one side of the pit 105 of a desirable mine site 101. For example, Figure 2 illustrates such a situation. The valuable material 102 is located to one side of the pit 105. In such a situation, it is not considered efficient to remove the waste material 104 from region 206, that is where the waste material is not located relatively close to the valuable material

102, but it is considered desirable to remove the waste material 104 from region 207, that is where it is located nearer to the valuable material 102. This then brings other considerations to the fore. For example, it would be desirable to determine the boundary between regions 206 and 207, so that not too much undesirable waste material is removed (region 206), yet enough is removed to ensure safety factors are considered, such as cave-ins, etc. This then leads to a further consideration of the need to design a 'pit' 105 with a relatively optimal design having consideration for the location of the valuable material, relative to the waste material and other issues, such as safety factors.

This further consideration has led to an analysis of pit design, and a technique of removing waste material and valuable material called 'pushbacks'. This technique is illustrated in Figure 3. Basically, the pit 105 is designed to an extent that the waste material 104 to be removed is minimised, but still enabling extraction of the valuable material 102. The technique uses 'blocks' 308 which represent smaller volumes of material. The area proximate the valuable material is divided into a number of blocks 308. It is then a matter of determining which blocks need to be removed to enable access to the valuable material 102. This determination of 'blocks 308', then gives rise to the design or extent of the pit 105.

Figure 3 represents the mine as a two dimensional area, however, it should be appreciated that the mine is a three dimensional area. Thus the blocks 308 to be removed are determined in phases, and cones, which represent more accurately a three dimensional 'volume' which volume will ultimately form the pit 105.

Furthermore, a number of prior art techniques are considered to take a relatively simple view of the problems confronted by the mine designer in a 'real world' mine situation. For example, the size, complexity, nature of blocks, grade and other engineering constraints and time taken to undertake a mining operation is often not fully taken into account in prior art techniques, leading to computational problems or errors in the mine design. Such errors can have significant financial and safety implications for the mine operator.

With regard to size, for example, prior art techniques fail to adequately take account of the size of a 'block'. Depending on the size of the overall project,

a 'block' may be quite large, taking some weeks, months or even years to mine. If this is the case, many assumptions made in prior art techniques fail to give sufficient accuracy for the modern day business environment.

It is desirable to provide an improved mine design.

Dynamic Programming Approach

With this in mind, consider the Lerchs-Grossman graph-theoretic algorithm for the ultimate pit problem (H. Lerchs & I. Grossman, "Optimum Design of Open-Pit Mines", Transactions CIM, 1965) which has been proved to give a relatively exact solution to the ultimate pit problem for an open-cut mine in three dimensions. Lerchs and Grossman also presents a dynamic programming approach to the problem in two dimensions, which has since been extended to three dimensions. However, solution of the three-dimensional graph theoretic algorithm is computationally inefficient in practical cases.

Linear Programming Approach

An alternative approach is to model the problem as a linear program (LP), as presented by Underwood and Tolwinski (R. Underwood & B. Tolwinski, "A mathematical programming viewpoint for solving the ultimate pit problem", EJOR, 1998). The availability of CPLEX (by Ilog, www.ilog.com) as a powerful LP solver motivates investigation of the LP approach to the ultimate pit problem.

The ultimate pit problem can be modelled as an integer program (IP), where a value of 1 is assigned to blocks included in the ultimate pit, and a value of 0 is assigned otherwise. The IP formulation for the problem is then as follows.

Let

$$x_i = \begin{cases} 1, & \text{if block } i \text{ is included in the ultimate pit} \\ 0, & \text{otherwise} \end{cases}$$

Then

$$\max \sum_i v_i x_i$$

$$s.t.$$

$$x_i \le x_j \qquad \forall j \in P(i)$$

$$x_i \in \{0,1\} \qquad \forall i \qquad \text{......equation 1}$$

where

v_i is the value assigned to block i

x_i is the decision variable that designates whether block i is included in the ultimate pit or not

P(i) is the set of predecessor blocks of block i.

One objective is to maximise the net value of the material removed from the pit. Consider that the only constraints are precedence constraints, which enforce the requirement of safe wall slopes in the mine. In fact, this IP formulation has the property of total unimodularity. That is, the solution of the LP relaxation of this formulation will be integral (i.e. a set of 0's and 1's). This is an extremely desirable property for an integer program. It allows the IP to be solved as an LP using the Simplex method. This leads to greatly increased solution efficiency in terms of both CPU time and memory requirements. The exact mathematical formulation of the linear programming approach to the ultimate pit problem is therefore

$$\max \sum_i v_i x_i$$

$$s.t.$$

$$x_i \le x_j \qquad \forall j \in P(i)$$

$$0 \le x_i \le 1 \qquad \forall i \qquad \text{......equation 2}$$

This is the ideal approach to solve the problem, and is considered to give the optimal solution in every case. Unfortunately, implementation of this exact formulation in CPLEX fails to solve for mining projects of realistic size. Since the optimisation is carried out at the block level, and there is a constraint for every precedence arc for each block, a very large number of constraints are applied. For example, if a mine has 198,917 blocks, and after CPLEX performs pre-

processing on the formulation, the resulting reduced LP still has 1,676,003 constraints. CPLEX attempts to solve this formulation using the dual simplex method, generally recognized as the most efficient method for solving linear programs of this size. However, in the case of the example mine, CPLEX was found to crash during the solution process due to the very large number of constraints. Inversion of a constraint matrix of this magnitude (as required for converting solutions obtained from the dual simplex method back into primal space) is considered to place too great a memory requirement on the system.

There still exists a need, however, to improve prior art techniques. Given that mining projects, on the whole, are relatively large scale operations, even small improvements in prior art techniques can represent millions of dollars in savings, and / or greater productivity and / or safety.

An object of the present invention is to determine which blocks of a mine pit provide a relative maximum net value of material, also having regard to practical limitations, such as slope constraints.

Any discussion of documents, devices, acts or knowledge in this specification is included to explain the context of the invention. It should not be taken as an admission that any of the material forms a part of the prior art base or the common general knowledge in the relevant art in Australia or elsewhere on or before the priority date of the disclosure and claims herein.

SUMMARY OF INVENTION

The present invention provides a method of determining a selected group of blocks of a mine pit which are capable of being mined, the method including the steps of selecting a plurality of blocks, and determining a relative value and constraints applicable to the selected blocks in accordance with any one of the equations 3, 4 or 9 as disclosed herein.

The present invention also provides the method as described above and including the further step of testing for violations.

The present invention also seeks to reiterate the selection and determination of value and constraints of blocks in order to obtain a group of blocks which have a relative optimal mining value.

Other aspects and preferred aspects are disclosed in the specification and/or defined in the appended claims, forming a part of the description of the invention.

In essence, the present invention, in various aspects, utilises aggregating algorithm(s) to determine a selected group of blocks which are to be mined, where the selection of blocks to be included into the group of blocks is made relative to value and constraints applicable to the blocks. The present invention, in another aspect further tests for violations, and iteratively recalculates until substantially all violations are removed. Given a block model of an ore body containing value-in-ground and designated slope constraints, the ultimate pit problem concerns the determination of the shape of the final pit of the mine. It is assumed that all the material can be removed at once. That is, the effect of time on the value of the ore body is not considered. In terms of mine scheduling, the ultimate pit can be used as the initial collection of blocks on which a scheduling algorithm is run. In this respect, the ultimate pit is the largest possible final pit that can be realised following scheduling of removal of the ore body. The case considered throughout this disclosure is that of base metals but also has application to blended products or stochastic elements of open-pit mining.

Throughout the specification, reference to block constraints equally implies reference to arc constraints. A block may also refer to a number of blocks.

DESCRIPTION OF DRAWINGS

Further disclosure, objects, advantages and aspects of the present application may be better understood by those skilled in the relevant art by reference to the following description of preferred embodiments taken in conjunction with the accompanying drawings, in which:

Figures 1 to 3 illustrate prior art mining techniques,

Figure 4 illustrates a comparison between outcomes of equations 2 and 4,

Figure 5 illustrates a vertical cross-section of a pit design using equation 2,

Figure 6 illustrates a vertical cross-section of a pit design using equation 4,

Figure 7 illustrates an example portion of a pit,

Figures 8 and 10 illustrate a plane view through a pit using the cutting plane formulation (equation 9), and

Figures 9 and 11 illustrate the same view as that of Figures 8 and 10 but for the use of the LP relaxation of the aggregated formulation (equation 4).

DETAILED DESCRIPTION

First aspect of invention

An approach in accordance with a first aspect of invention is to aggregate the precedence constraints as follows:

$$\max \sum_i v_i x_i$$

$$s.t.$$

$$n_i x_i \leq \sum_{j \in P(i)} x_j$$

$$x_i \in \{0,1\} \quad \forall i$$

$$\text{where} \quad n_i = |P(i)|$$

..........equation 3

In this first aspect approach, the number of constraints is reduced to one for every block below the surface (there are no precedence constraints for the blocks on the top bench of the pit). In this case each constraint enforces the rule that a block can only be extracted if all of its predecessor blocks are extracted. However, the total unimodularity property of the exact (disaggregated) formulation is not preserved in this first approach formulation. Hence, the integrality constraints on the decision variables must be enforced. Equation 3 manifests therefore as an integer program, and must be solved using the method of branch-and-bound, rather than the Simplex method. This solution method takes a relatively long time in terms of computation time and can also require a relatively large amount of memory for storage of the decision tree. In particular, obtaining the truly optimal solution (as opposed to a solution within a specified percentage of the optimal solution) may take a relatively long time.

When the aggregated formulation (equation 3) is LP-relaxed and solved in CPLEX, the decision variables may take fractional values, and the outcome is expressed in equation 4 following:

$$\max \sum_i v_i x_i$$

$$s.t.$$

$$n_i x_i \le \sum_{j \in P(i)} x_j$$

$$0 \le x\_i \le 1 \forall i$$

$$\text{where } n_i = |P(i)|$$

equation 4

Consider the case of a relatively small first example of a mine (16,049 blocks) that is provided as an example with the Whittle software package (by Whittle Pty Ltd, www,whittle.com.au). Figure 4 shows the view from above of a comparison of the optimal solutions found by the exact formulation (equation 2) and the LP relaxation of the aggregated formulation (equation 4). The blocks 10 are those that are set to 1 by both the exact formulation (equation 2) and the aggregated formulation (equation 3). The blocks 11 around the outside of this pit are those blocks which are included (set to 1) in the ultimate pit found by the exact formulation (equation 2), but are not included (set to 0) in the solution found by the LP relaxation of the aggregated formulation (equation 4). It is evident that there are a number of blocks that are included in the true ultimate pit that are not included by the LP relaxation of the aggregated formulation (equation 4). The blocks 12 are waste.

A comparison of a vertical cross-section of the pit design using the exact formulation (equation 2) and the LP relaxation of the aggregated formulation (equation 4) for this first mine example is illustrated in Figure 5 when compared with Figure 6.

Figure 5 shows a plane through the example pit from the view of the solution using the exact formulation (equation 2). The area 20 is the ultimate pit and the area 21 is waste. Referring to Table 1, below, the total value of this pit is found to be $1.43885E+09, and CPLEX requires 29.042 seconds to obtain this solution.

Figure 6 shows the equivalent view when the LP relaxation of the aggregated formulation (equation 4) for the ultimate pit is used. The area 20 is blocks set to 1, area 21 is waste (blocks set to 0) and area 22 is material which

may be further interrogated in order to decide whether it is included (or not) in the ultimate pit (set to a value between 0 and 1). The total value of this pit is found to be $1.54268E+09, and found in a CPU time of 0.992 seconds. Note that the solution of the aggregated formulation (equation 3) (where integrality constraints are imposed on the decision variables) gives a total value of the ultimate pit to be $1.43591E+09 (using a branch-and-bound stopping criteria of 1% from optimal), which is similar to the value as that given by equation 2, and a CPU time of 1675.18 seconds was required to obtain this solution.

First example mine	**Total Blocks**	**16049**
Formulation		
Exact LG (equation 2)		
Total Number of Precedence Constraints	264859	
Total Value	1.43885E+09	
CPU Time (Seconds)	29.402	
No. Blocks in Ultimate Pit	9402	
% of Total Blocks	58.58	
Aggregated LG (equation 3)		
(IP)		
Total Number of Precedence Constraints	14077	
Total Value	1.43591E+09	
CPU Time (Seconds)	1675.18	
No. Blocks in Ultimate Pit	9670	
% of Total Blocks	60.25	
Final Gap (from optimal)	0.46%	
Aggregated LG (equation 4)		
(LP relaxation)		
Total Number of Precedence Constraints	14077	
Total Value	1.54268E+09	
CPU Time (Seconds)	0.992	
No. Blocks in Ultimate Pit	7949	
% of Total Blocks	49.53	
Aggregated LG (Cutting Plane) (equation 9, below)		
(LP relaxation + add single block constraints)		
Total Number of Precedence Constraints	34819	
Total Value	1.43885E+09	
CPU Time (Seconds)	976.565	

No. Blocks in Ultimate Pit	9402	
% of Total Blocks	58.58	
Number of Iterations	9	

Table 1: Summary of results for first mine example.

It is evident that CPLEX, when using this relaxed aggregated formulation for the problem, provides a relatively higher valued ultimate pit to be found, but does so in a relatively shorter time. This relatively higher value results, in part, from a relaxation of the predecessor constraints, thus allowing a fraction of a block to be taken even when all of its predecessor blocks have not been taken.

By way of illustration of the reason for finding a relatively higher pit value using equation 4, consider the situation shown in Figure 7. The number within each block represents the value assigned to the decision variable (x_i) for that block by the LP relaxation of the aggregated formulation (equation 4).

In the case illustrated in Figure 7, Blocks 2 and 3 are predecessors of Block 1. Block 1 is represented by x_1, block 2 by x_2 and block 3 by x_3 in the equations below. In the exact formulation (equation 2), the constraints for this situation illustrated are

$$x_1 \le x_2$$
$$x_1 \le x_3$$

......equation 5

The solution given (x1 = 0.5, x2 = 0, x3 = 1) is infeasible for the exact formulation (equation 2), since

$$x_1 = 0.5 > x_2 = 0$$

equation 6

However, in the LP relaxation of the aggregated formulation (equation 4), the relevant constraint is

$$2x_1 \le x_2 + x_3$$

......equation 7

In this case the solution from Figure 7 is considered feasible (since 2 x 0.5 = 1 <= 0 + 1 = 1).

$$2 \times \frac{1}{2} \le 0 + 1$$

......equation 8

Hence if Blocks 1 and 3 were ore blocks and had positive value, while Block 2 was a waste block with negative value, the LP relaxation of the

aggregated formulation (equation 4) can take all of Block 3 and 0.5 of Block 1 without incurring the penalty of taking the negative valued Block 2. Hence the aggregated formulation (equation 4) can take fractions of positive blocks that otherwise would not have been taken in the exact formulation (equation 2). This leads to a solution of greater value than in the disaggregated case.

Second aspect of invention

The LP relaxation of the aggregated formulation (equation 4) can be modified to overcome this solution of artificially greater value. The result is equation 9 below, namely:

$$\max \sum_i v_i x_i$$

$$s.t.$$

$$n_i x_i \le \sum_{j \in P(i)} x_j$$

$$0 \le x\_i \le 1 \, \forall i$$

$$\text{where } n_i = |P(i)|$$

loop over all arcs

{ if $i \rightarrow j$, and $x_i > x_j$ in solution, then add the constraint $x_i \le x_j$ }

.........equation 9

This approach as expressed by equation 9 is considered a second aspect of invention termed a 'cutting plane method'. In this second aspect, an initial (reduced) problem is solved to give an upper bound on the optimal value, and then any constraints from the overall (Master) problem that are violated by this solution are added, and the problem is re-solved. This is repeated until substantially no constraints from the Master problem are found to be violated. In this second aspect, the linear program for the aggregated formulation (equation 4) is run and a solution, call it $\hat{x}$ is obtained. Each element of the vector $\hat{x}$ represents the value (possibly fractional) assigned to each block. Within $\hat{x}$ there will be instances of pairs of individual blocks where the constraint that the successor block cannot be taken until the entire predecessor block has been taken (from the exact formulation) is violated. For example, in Figure 7, the constraint in the exact formulation that block 1 is assigned an i value of 0.5 and j is assigned a value of 0

$x_1 \leq x_2$ .....:.equation 10

is violated, since x1 = 0.5 and x2 = 0.

Thus, in the case of Figure 7, i has a value greater than j and the constraint is added and the solution re-run. The result will be the violation posed by Figure 7 as far as blocks 1 and 2, will be removed. Some individual block constraints can be added to the LP relaxation of the aggregated formulation (equation 4) to make it feasible for the ultimate pit problem. It is possible to perform the following iteration.

For each element of $\hat{x}$, compare its value with that of each of its predecessor blocks in turn. Whenever there is a situation where the successor block has a greater value than the predecessor block, add the relative single block constraint to the formulation. For example, in the situation from Figure 7, the constraint

$x_1 \leq x_2$

will be added to the LP relaxation of the aggregated formulation (equation 4). After checking the relationship for all pairs of predecessors, re-solve the problem, subject to the aggregated constraints as well as the added single block precedence constraints. Again, the solution may be infeasible, so the process may have to be repeated. This process should be repeated until the step of checking single block dependencies reveals that substantially no single block precedence relationships are violated. The solution at this point has been found to be the same as the optimal solution, found by solving the exact formulation (equation 2).

It is considered that the number of constraints needed to obtain the solution using this second aspect approach is significantly less than the number used in the disaggregated formulation. Since the initial aggregated solution gives a reasonable approximation to the ultimate pit, it has been found that only a small percentage of the total number of single block precedence constraints for the problem should need to be added to the formulation. In this way, the computational requirement in terms of memory (storage and manipulation of the constraint matrix) to find the optimal solution should be significantly reduced. However, the cost of this approach is that the process of checking and

identification of violated constraints will require more time than the prior art method of equation 2. When equation 9 is applied to the first mine example referred to above, this second approach found the total value of the pit to be \$1.43885E+09, the same as the solution to the problem using the disaggregated formulation (equation 2). The computation time required to achieve this second approach was 976.565 seconds.

A brief comparison of these two methods for the ultimate pit problem at the first example mine is given in Table 1, above.

Third aspect of Invention

It is evident that the trade off between the prior art approach and the approaches of the first and second aspects is time against memory, as illustrated in Table 1, above). The exact formulation (equation 2) finds the optimal solution in 29.402 seconds, while the cutting plane formulation (equation 9) takes 976.565 seconds to find the optimal solution. This is due, in part, to the fact that the cutting plane formulation re-solves a large LP a number of times in the process of solving the problem. In addition, the process of searching through and checking the entire arcs file (which is completed as a part of each iteration) takes a significant amount of time. However, the exact formulation (equation 2) solves a model with 264,859 precedence constraints (requiring a significant amount of memory), compared with 34,819 precedence constraints in the cutting plane formulation (equation 5). This is a decrease of 87%. It is expected that the number of constraints in the model is proportional to the memory required to store and solve the problem, in particular, to perform the inversion on the final constraint matrix once the optimal solution has been found. Thus, advantageously, a solution of the cutting plane formulation (equation 9) may be possible in cases where CPLEX runs out of memory when trying to solve the exact formulation (equation 2).

In a second example mine, which has 38,612 blocks, the same approach was taken to that above, with similar results, as shown in Table 2.

Example Mine 2	Total Blocks	38612
Formulation		
Exact LG (equation 2)		
Total Number of Precedence Constraints	1045428	
Total Value	1.87064e+009	
CPU Time (Seconds)	223.762	
No. Blocks in Ultimate Pit	33339	
% of Total Blocks	86.34	
Aggregated LG (Cutting Plane) (equation 9)		
(LP relaxation + add arc or single block constraints)		
Total Number of Precedence Constraints	159832	
Total Value	1.87064E+09	
CPU Time (Seconds)	12354.3	
No. Blocks in Ultimate Pit	33339	
% of Total Blocks	86.34	
Number of Iterations	6	

Table 2: Summary of results for second mine example.

In particular, referring to Table 2 above, the exact formulation (equation 2) contains 1,045,428 constraints, while the final model following implementation of the cutting plane algorithm (equation 9) requires only 159,832 constraints.

However, the cutting plane method (equation 9) takes 12,354.3 seconds to find the solution, while the exact formulation (equation 2) requires 223.762 seconds of CPU time.

Further testing of the alternative mixed integer program approaches to the pit design was carried out on a third mine example, as detailed in Table 3 below. The block model for the third mine example contains 198,917 blocks.

Initially, the exact formulation (equation 2) was trailed. This resulted in CPLEX attempting to solve a linear program with 3,526,057 single block constraints. The size of this constraint matrix caused CPLEX to run out of memory when trying to apply the dual simplex algorithm to solve the problem. Thus, the exact solution to the pit design in the case of this third mine example is unable to be determined by this approach.

The aggregate formulation (equation 3) was next trailed. This resulted in 188,082 constraints, a value of $3.34125E+09, and a CPU time of 33298.5 seconds.

The next trail was to run the LP relaxation of the aggregated formulation (equation 4). It is expected that the solution to this problem will give an upper bound on the optimal value of the ultimate pit, as was described above. This is due to the fact that CPLEX includes fractions of blocks without necessarily taking their entire precedence set. In this trail, the model had 188,082 constraints. The optimal solution was found to have a value of $3.40296E+09, and this was found in 12.989 seconds of CPU time.

example Mine 3	**Total Blocks**	198917
Exact LG (equation 2)		
Total Number of Precedence Constraints	3526057	
Total Value		
CPU Time (Seconds)	out of memory	
No. Blocks in Ultimate Pit		
% of Total Blocks		
Aggregated LG (equation 3)		
(IP)		
Total Number of Precedence Constraints	188082	
Total Value	3.34125E+09	
CPU Time (Seconds)	33298.5	
No. Blocks in Ultimate Pit	97221	
% of Total Blocks	48.88	
Final Gap (from optimal)	0.99%	
Aggregated LG (equation 4)		
(LP relaxation)		
Total Number of Precedence Constraints	188082	
Total Value	3.40296E+09	
CPU Time (Seconds)	12.989	
No. Blocks in Ultimate Pit	91522	
% of Total Blocks	46.01	
Aggregated LG (Cutting Plane) (equation 9)		
(LP relaxation + add single block or arc constraints)		
Total Number of Precedence Constraints	285598	
Total Value	3.37223E+09	
CPU Time (Seconds)	19703.8	
No. Blocks in Ultimate Pit	98845	
% of Total Blocks	49.69	
Number of Iterations	4	

Table 3: Summary of results for third mine example.

The cutting plane formulation (equation 9) was also trailed on this example third mine. This is the method where the solution to the LP relaxation of the

aggregated formulation is used as a starting solution, and then violated single block constraints are added to the model and then again resolved. This process is repeated until no more single block constraints are violated, and thus the solution is similar to that for the exact formulation. The solution to this equation 9 is considered to be the correct solution to the problem. When equation 9 was run, it was found that CPLEX was able to handle the size of the problem, and the exact ultimate pit was found. The solution contained 285,598 constraints, a reduction of 92% on the exact formulation. The optimal value of the pit design was found to be $3.37223E+09, and the CPU time required to find this solution was 19703.8 seconds.

Thus the cutting plane algorithm (equation 9) has been found to provide an improved solution within the memory limits of a practical implementation of the present invention, using computers and / or computer modelling, where the exact formulation (equation 2) could not. Again, the saving in memory is offset by a longer computation time.

As in the case of the first mine example, a comparison of a vertical cross-section of the solution to the ultimate pit problem using the cutting plane formulation and the LP relaxation of the aggregated formulation for the third mine example is illustrated in the Figures. Figures 8 and 10 show a plane view through the pit using the cutting plane formulation (equation 9). The area 20 is the ultimate pit and the area 21 is waste. Figures 9 and 11, on the other hand, show the same view, but for the LP relaxation of the aggregated (equation 4). Again, areas 20 are the pit and areas 21 are waste. Again, it is evident that the LP relaxation of the aggregated (equation 4) takes fractions of blocks that are infeasible for the exact formulation.

This result is considered to confirm that solution of the cutting plane formulation (equation 9) may be possible in cases where CPLEX runs out of memory when trying to solve the exact formulation (equation 2).

A summary of the results for the third mine example is found in Table 3.

A Fourth Aspect of Invention

Variations On The Cutting Plane Method

First variation

Since it was found that adding all violated constraints at once causes additional loading on the cutting plane approach (equation 9), due to the very large number of constraints added by the first iteration, one variation of the cutting plane method is to add the constraints incrementally. Initially, the effect of adding the most violated constraints first, and then re-solving the formulation was investigated. This method was thoroughly tested on the first mine example. The approach taken was as follows. At each iteration of the method, a lower bound on the size of the violation of the single block constraint was specified (e.g. 0.5, 0.6, ...). For example, Figure 7 illustrates violations for each block. In this example Figure 7, the violation = $x_i - x_j$, and so the 'size' of the violation is 0.5 – 0 = 0.5. Constraints that were violated by an amount greater than this tolerance were added to the formulation, and the problem was re-solved. However, using this approach the optimisation process completed before the optimal solution was found. This occurs because this method of adding constraints does not identify and add all single block constraints that are violated, only those that are violated by more than a certain amount. In this way, not all of the necessary single block constraints are added to the formulation, and the truly optimal solution is not reached. To alleviate this problem, violation(s) greater than a selected lower bound is added to at least the first iteration. This approach enables an optimal solution is still obtained.

Second variation

Another approach is to add the most violated constraints, but to decrease the amount of violation required at each iteration until a certain number of constraints have been added. For example, it may be designated that a minimum of 5000 constraints should be added at each iteration. Say the initial violation parameter is set to 0.6 (that is, only single block constraints that are violated by 0.6 or more are added to the formulation). It may be the case that 1200 constraints are added. Then, before re-solving the formulation, the violation parameter could be decreased to 0.5. This may result in a further 3000 constraints being added to the model. Since there are still less than 5000

constraints added, the violation parameter is further decreased to 0.4, and more single block constraints are added. This may result in 2000 constraints being added to the formulation, and the problem is now re-solved since the minimum of 5000 constraints has been reached. The process is then repeated until the optimal solution is obtained.

Third variation

Alternatively, the tolerance could be reduced on a smaller incremental level (say 0.01 at a time instead of 0.1) in an attempt to reduce the size of the overshoot on the number of constraints added compared with the prescribed minimum number of constraints.

Fourth variation

A further alternative is simply to add a specified number of constraints to the model before the formulation is re-solved. In any approach where a minimum number of constraints are added, the determination of the appropriate number of constraints to add at each iteration is a non-trivial matter. This element of the problem may itself require optimisation. It is expected that the maximum size of the problem that is able to be stored in memory and handled by CPLEX will affect this value. Consideration of this fact may allow a test to be built in to the program for solving the ultimate pit problem. The form of the test procedure could proceed as follows. If the size of the constraint matrix following the first iteration is less than the maximum size able to be solved by CPLEX, (with a margin to allow more constraints to be added in subsequent iterations based on the general proportion of constraints added after the initial loop – it appears that approximately 90% of the constraints that are required are added in the first loop), take the path of adding all violated constraints. If the size of the constraint matrix following the first iteration is greater than the maximum able to be solved, restart the iteration process using one of the alternative constraint-adding processes described above.

The approaches described above were tested on the first mine example above. In this case, the approach that performed the best was to add single block constraints that were violated by more than 0.6 in the first 5 loops, and in subsequent loops, add all violated constraints. This approach found the optimal solution in 2152.24 seconds. This was significantly longer than the standard

cutting plane procedure, which required 976.565 seconds (compare with statement below).

Fifth variation

Another approach for adding constraints incrementally takes advantage of the specific geometry of the mine. In this case, a vector containing the z coordinate (or "height") for each block is stored. Using this information, violated single block constraints are added from the largest z coordinate (corresponding to the top of the pit) down, decreasing by block height, in each loop. The constraint adding process stops either once a specified number of constraints have been added, or after a specified number of z coordinates have been descended. By adding violated single block constraints from the largest z coordinate down, it is hoped that the subsequent optimisation steps will force more single block constraints from lower in the pit to be satisfied before they need to be explicitly added to the formulation in a cutting plane iteration. That is, once decisions regarding the uppermost benches of the pit have been made, the precedence constraints within the formulation could force these decisions to propagate down the pit. Subsequently, less single block constraints may need to be added through the cutting plane iterations before the problem is solved to optimality.

This approach was particularly effective in the case of the third mine example. The optimal solution to the problem was found in 2664.11 seconds when constraints were added from the top z coordinate down in each iteration, with ten z coordinates descended in each iteration. This compares very favourably with the standard cutting plane formulation, which requires 19,703.8 seconds to find the optimal solution.

While this invention has been described in connection with specific embodiments thereof, it will be understood that it is capable of further modification(s). This application is intended to cover any variations uses or adaptations of the invention following in general, the principles of the invention and including such departures from the present disclosure as come within known or customary practice within the art to which the invention pertains and as may be applied to the essential features hereinbefore set forth.

As the present invention may be embodied in several forms without departing from the spirit of the essential characteristics of the invention, it should be understood that the above described embodiments are not to limit the present invention unless otherwise specified, but rather should be construed broadly within the spirit and scope of the invention as defined in the appended claims. Various modifications and equivalent arrangements are intended to be included within the spirit and scope of the invention and appended claims. Therefore, the specific embodiments are to be understood to be illustrative of the many ways in which the principles of the present invention may be practiced. In the following claims, means-plus-function clauses are intended to cover structures as performing the defined function and not only structural equivalents, but also equivalent structures. For example, although a nail and a screw may not be structural equivalents in that a nail employs a cylindrical surface to secure wooden parts together, whereas a screw employs a helical surface to secure wooden parts together, in the environment of fastening wooden parts, a nail and a screw are equivalent structures.

THE CLAIMS DEFINING THE INVENTION ARE AS FOLLOWS:

1. A method of determining a selected group of blocks of a mine pit which are capable of being mined, the method including the step of

selecting a first plurality of blocks, and

determining a relative value and constraints applicable to the selected first plurality of blocks in accordance with:

$$\max \sum_i v_i x_i$$

$$s.t.$$

$$n_i x_i \leq \sum_{j \in P(i)} x_j$$

$$x_i \in \{0,1\} \quad \forall i$$

$$\text{where} \quad n_i = |P(i)|$$

..........equation 3

2. A method of determining a selected group of blocks of a mine pit which are capable of being mined, the method including the step of

selecting a first plurality of blocks, and

determining a relative value and constraints applicable to the selected first plurality of blocks in accordance with:

$$\max \sum_i v_i x_i$$

$$s.t.$$

$$n_i x_i \leq \sum_{j \in P(i)} x_j$$

0 ≤ x_i ≤ 1 ∀i

$$\text{where}\, n_i = |P(i)|$$

.........equation 4

3. A method of determining a selected group of blocks of a mine pit which are capable of being mined, the method including the step of

selecting a first plurality of blocks, and

determining a relative value and constraints applicable to the selected first plurality of blocks in accordance with:

$$\max \sum_i v_i x_i$$

$s.t.$

$$n_i x_i \leq \sum_{j \in P(i)} x_j$$

$$0 \leq x\_i \leq 1 \ \forall i$$

where $n_i = |P(i)|$

loop over all arcs

{ if $i \rightarrow j$, and $xi > xj$ in solution, then add the constraint $xi \leq xj$ }

.........equation 9

4. A method as claimed in claim3, further including the step of re-testing for violations.

5. A method as claimed in any one of claims 1 to 4, further including the further step of:

selecting a second plurality of blocks,

determining a relative value and constraints applicable to the selected second plurality of blocks in accordance with any one of equations 3, 4 or 9, and

determining whether the first or second plurality of block have a greater value.

6. A method as claimed in any one of claims 1 to 4, further including the step of repeating the further steps as claimed in claim 5 until substantially no constraints from the problem are found to be violated.

7. A method as claimed in claim 4, wherein a violation greater than or equal to a selected lower bound is added in the first iteration.

8. A method as claimed in claim 7, wherein, on subsequent iteration(s), a constraint with a reduced lower bound is added.

9. A method as claimed in claim 4, wherein a tolerance value is used to determine the number of constraints.

10. A method as claimed in claim 4 , wherein the number of constraints added is determined in accordance with:

a. If the number of constraints does not exceed a memory limit, then add substantially all the constraints, or

b. If the number of constraints exceeds a memory limit, then add some (only) of the constraints.

11. A method as claimed in claim 4, wherein the constraint added is in accordance with a z co-ordinate.

12. A method as claimed in claim 11, wherein the constraint added is in accordance with the largest z coordinate first.

13. A method of determining a selected group of blocks of a mine pit which are capable of being mined, the method including the step of

selecting a first plurality of blocks, and

determining a relative value and constraints applicable to the selected first plurality of blocks in accordance with:

a CPLEX method as defined by equation 2, disclosed herein, and

if the CPLEX method times out, redetermining the relative dimension and / or size of a mine pit using the method as claimed in any one of claims 1 to 12.

14. Apparatus adapted to determining a selected group of blocks of a mine pit which are capable of being mined, the apparatus including:

processor means adapted to operate in accordance with a predetermined instruction set, said apparatus, in conjunction with said instruction set, being adapted to perform the method as claimed in any one of claims 1 to 13.

15. A computer program product including:

a computer usable medium having computer readable program code and computer readable system code embodied on said medium for determining the relative dimension and / or size of a mine pit using a data processing system, said computer program product including:

computer readable code within said computer usable medium for determining a selection of blocks of a mine pit which are capable of being mined in accordance with the method as claimed in any one of claims 1 to 13.

DATED this 5th day of March 2003

By our Attorneys:

SMOORENBURG PATENT & TRADE MARK ATTORNEYS
PO BOX 9,
KANGAROO GROUND, VICTORIA, 3097
AUSTRALIA

FIGURE 1
(prior art)

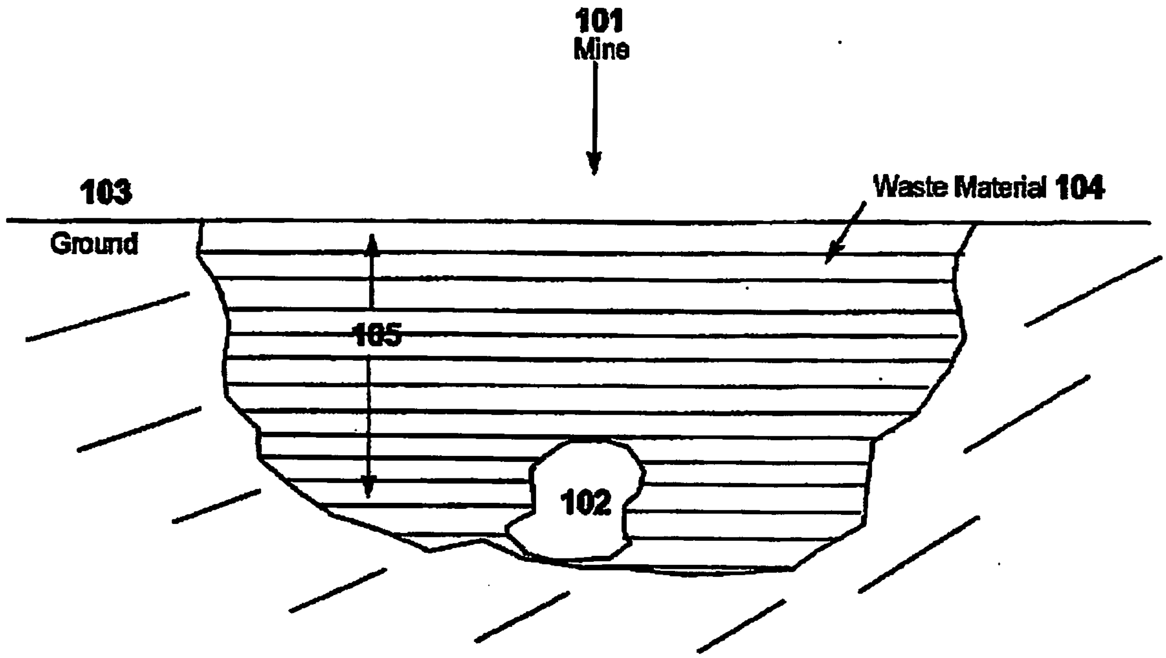


FIGURE 2
prior art

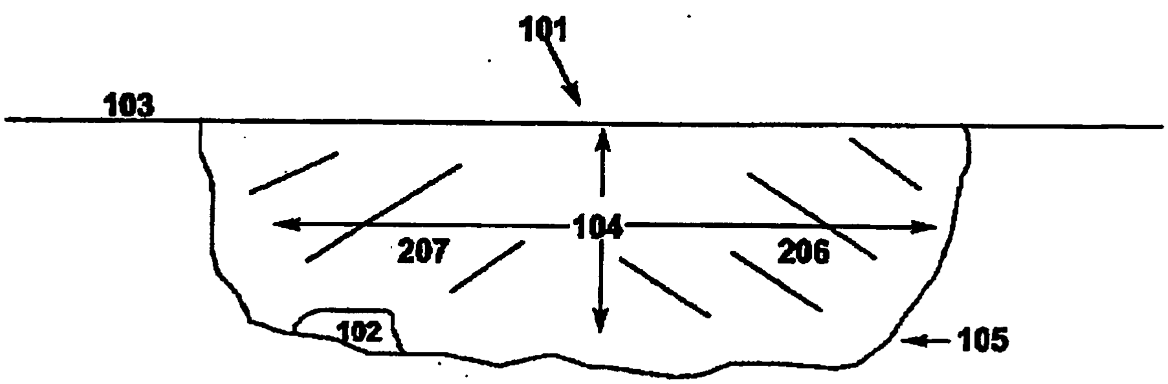
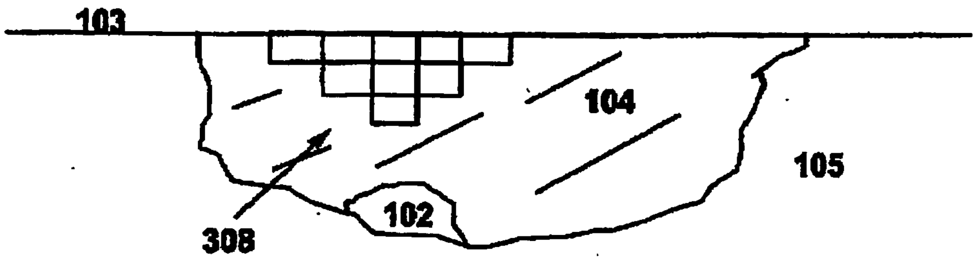


FIGURE 3
prior art



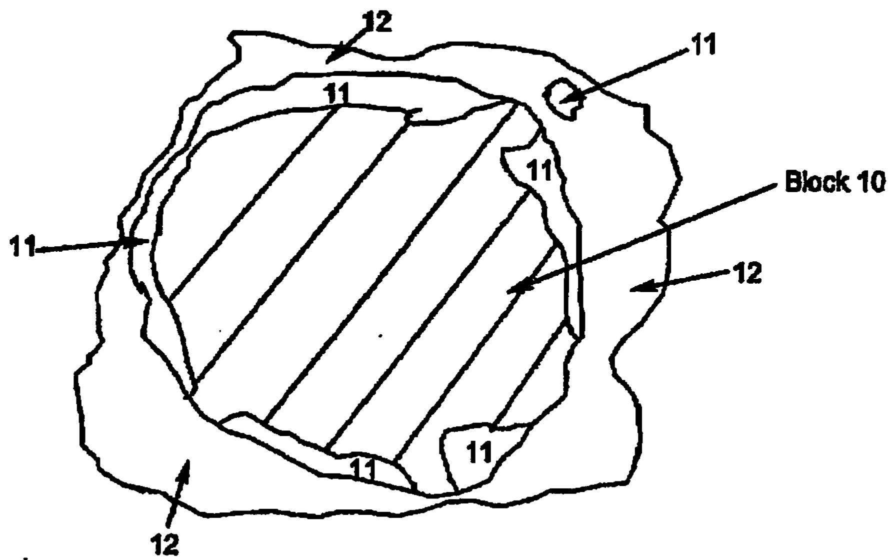


FIGURE 4

View from above identifying blocks included in the true ultimate pit but omitted by the LP relaxation of the aggregated formulation

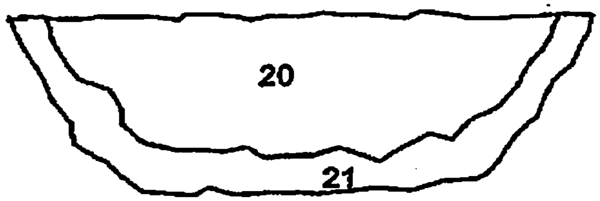


Figure 5

Vertical cross-section of the exact ultimate pit found for first mine example

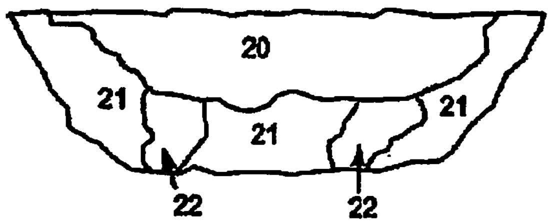


Figure 6

Vertical cross-section of the ultimate pit found using the LP relaxation of the aggregared formualtion of the first mine example

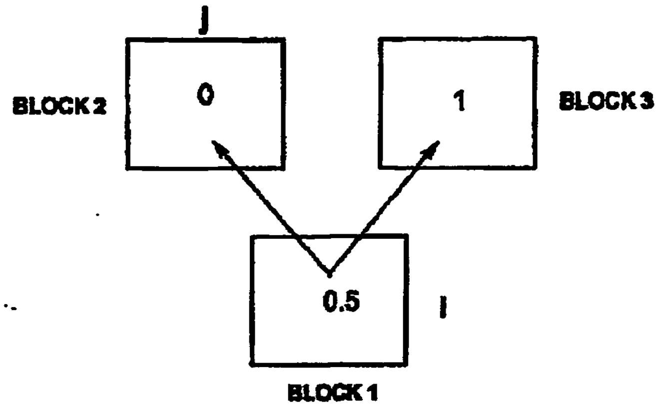


FIGURE 7

Example of feasible solution for LP relaxation of aggregared formulation that is infeasible for the exact formulation

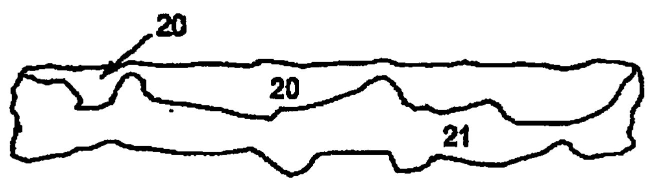


FIGURE 8

Vertical cross-section of the exact ultimate pit found for the third mine example

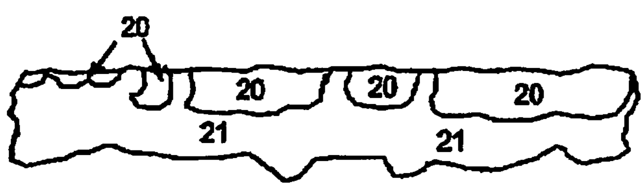


FIGURE 9

Vertical cross-section of the ultimate pit found using the LP relaxation of the aggregated formulation for the third mine example

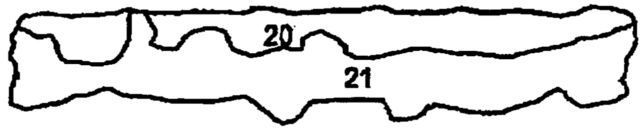


FIGURE 10

Vertical cross-section of the exact ultimate pit found for the thrid mine example

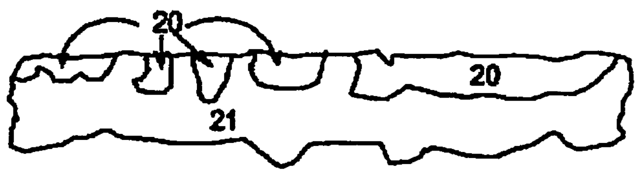


FIGURE 11

Vertical cross-section of the ultimate pit found using the LP relaxation of the aggregated formulation for the third mind example